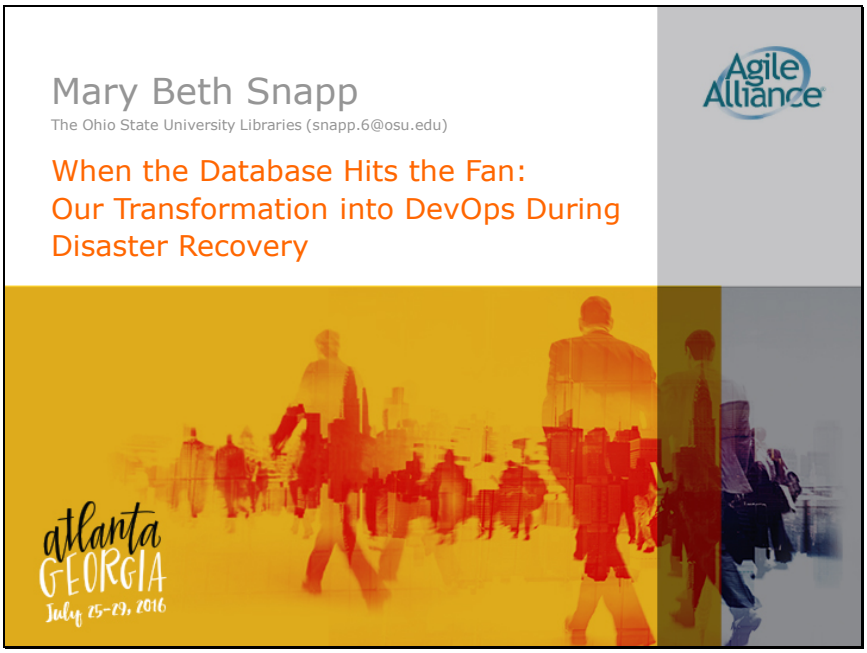
Slide 1



Mary Beth Snapp (snapp.6@osu.edu)
Head of Applications Development & Support
Information Technology Division
Ohio State University Libraries
Presented at Agile 2016, Atlanta, Georgia, July 26, 2016
https://agile2016.sched.com/event/6edn/when-the-database-hits-the-fan-our-transformation-into-devops-during-disaster-recovery-mary-beth-snapp

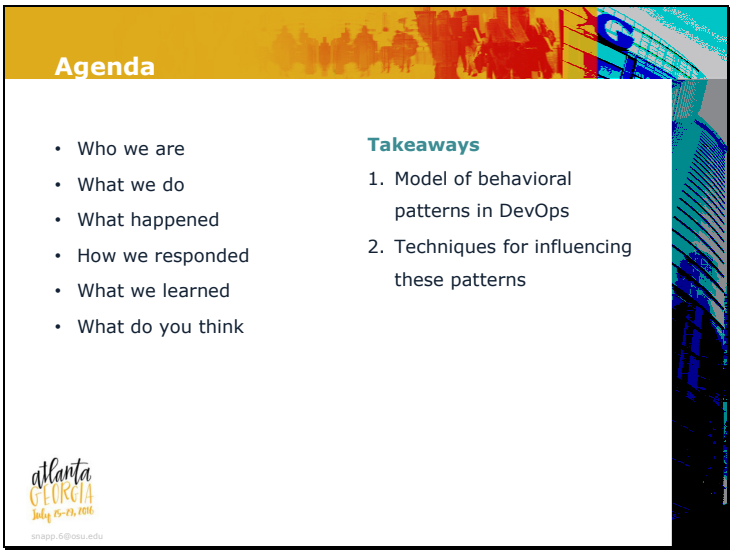
Slide 2

I'm a huge fan of Simon Sinek, and this is a quote from his 2014 TEDTalk that perfectly summarizes the theme of my presentation: how external pressures can bring people together.

Slide 3

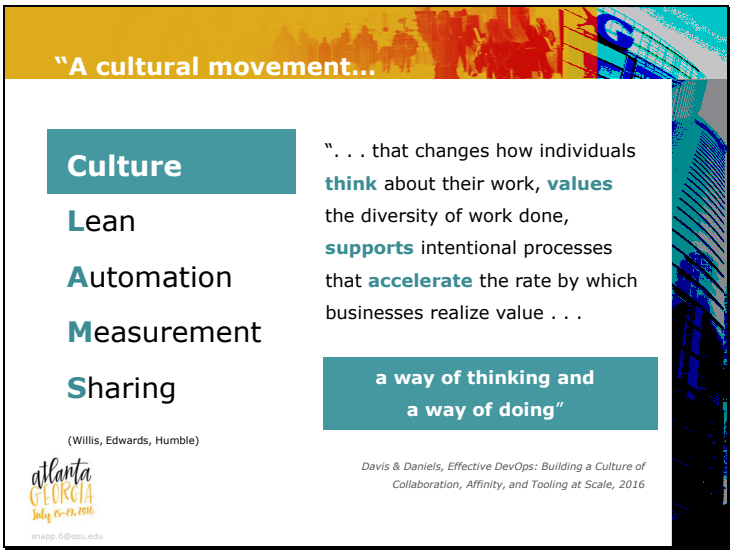
I've tweaked this quote for our particular circumstances which I'm sure many of you can relate to as IT professionals. I will talk today about how a disaster brought us together as a team and moved us forward in our transformation to DevOps.

Slide 4



I'll start by providing some context to the incident: who we are and what we do. I would like to spend the majority of the time talking about how we responded to the disaster. Then open it up for discussion so we can learn from one another.
There are 2 takeaways:
1) I'll present a model to make sense of what happened: that summarizes the team's behavioral patterns before, during and after the disaster.
2) I will offer a few practical techniques for influencing behavior patterns into a more productive DevOps culture.
I would like to suggest that paying attention to these patterns may provide you as a manager, some insight into the nature of the relationship between devs and sysadmins and how well they would work together in a highly pressurized situation, such as disaster recovery.

Slide 5

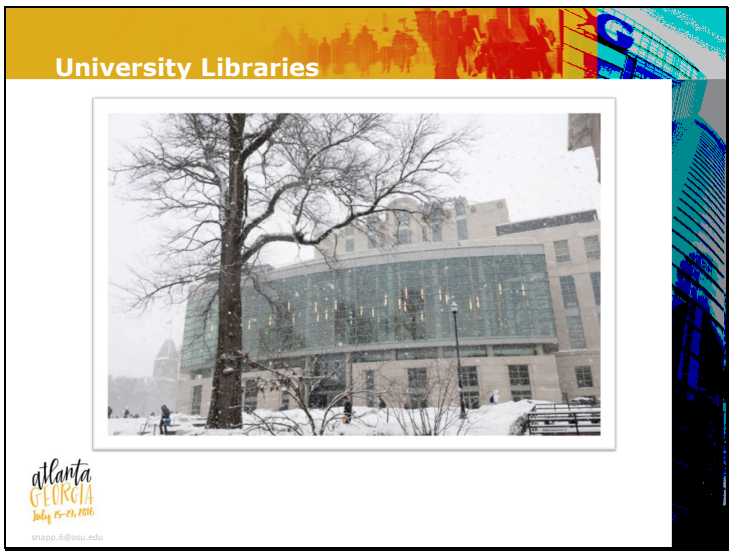


There is a lot of confusion around what DevOps really means with some very squishy and unsatisfying definitions. Reading about Continuous Delivery at Amazon and Etsy is how I was introduced to the idea of DevOps. The number and frequency of commits at these companies are somewhat intimidating for a small team in a non-profit organization. If that's DevOps, then that's not attainable for our team. So I'm in support of being careful to distinguish among tools, processes, outcomes, and culture, with continuous delivery as one possible outcome. I'm a particular fan of the CLAMS acronym, and today, I would like to focus on DevOps as culture--values, norms, relationships, ways of thinking, ways of behaving--that enables development teams and operations teams to work together toward a common goal.

Slide 6

Before I review the specifics of the disaster, I would like to provide some context for the project I will discuss today. As I mentioned I work at Ohio State in the Libraries, so our team is an embedded IT unit that operates relatively independently from the central IT of the university. The university is the 2nd largest public university in the US with 52,000 undergrads, 10,000 graduate students and 7,000 faculty. The library serves these students and researchers with over 9 million cataloged holdings at 15 locations.

Slide 7



Here is a winter picture of the library for those of you who are suffering in the Atlanta heat.

Slide 8

The library has 9 Special Collections.

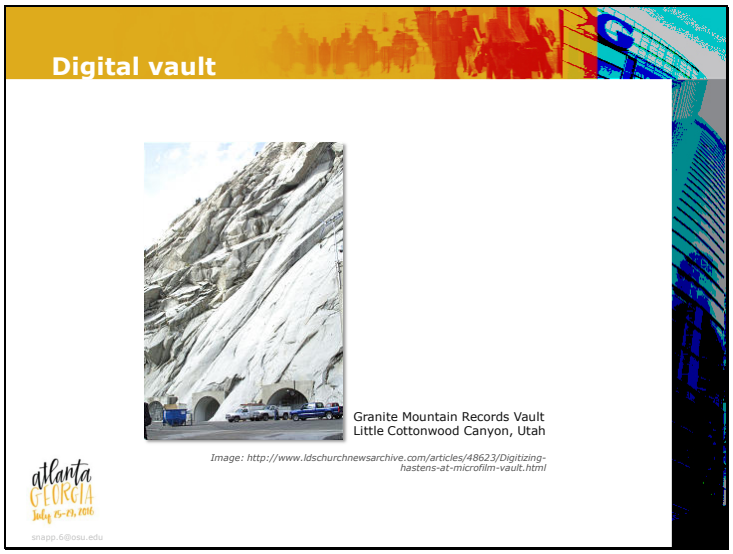
Slide 9

There are very old, fragile, and irreplaceable materials such as the King James Bible of 1611.

Slide 10

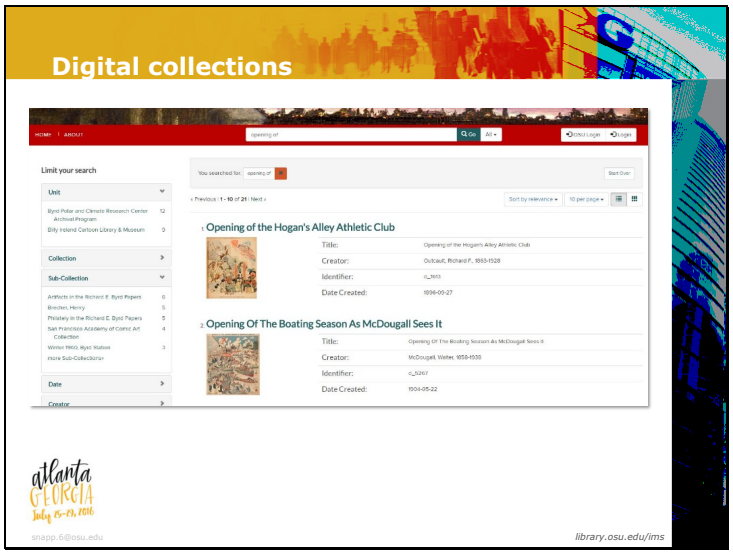
Our primary stakeholders for the project that I will talk about today is The Billy Ireland Cartoon Library & Museum—the world's largest collection of cartoons and comics. They have 300,000 original cartoons on site plus 2.5 million comic strip clippings. Researchers need to travel to Columbus to study these materials, by appointment only. That's a lot of paper hidden away from the public.

Slide 11



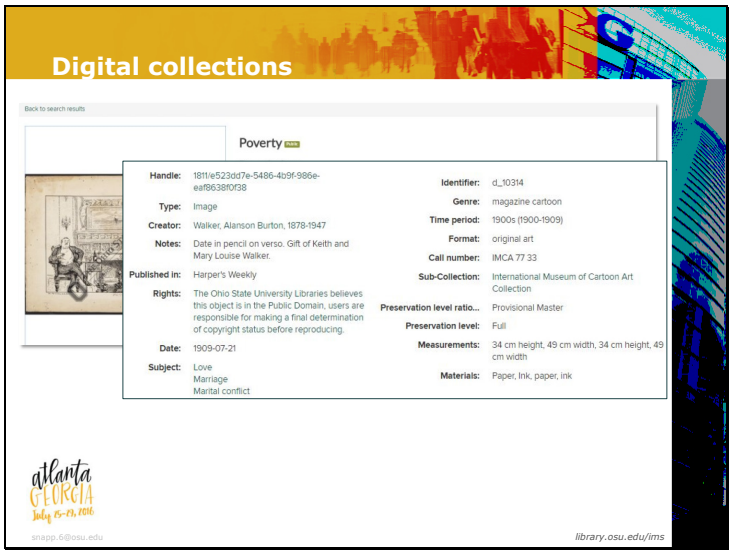
Our project is about building a digital vault for special collections called IMS. Digital vault means an archival, preservation-level repository for master objects. To a librarian, curator, or archivist, the expectation is that these digital images need to live on, in perpetuity, forever, and you should be able to find them among the ashes of the scorched earth. But our project is not just about preserving digital copies of unique materials, it's also about giving access to the public.

Slide 12



Our website, Image Management System, or IMS, is the front-end to the preservation repository. There are about 32,000 images in the repository now, and we have 2+ million more to ingest. That's just images. Eventually there will be audio, archival docs and manuscripts. We use the open source Fedora/Hydra stack which is a specialized repository platform mostly used by cultural heritage institutions, such as libraries.

Slide 13



Each item includes detailed metadata added by the curator. This is an example of digitized original art published in Harper's Weekly in 1909.

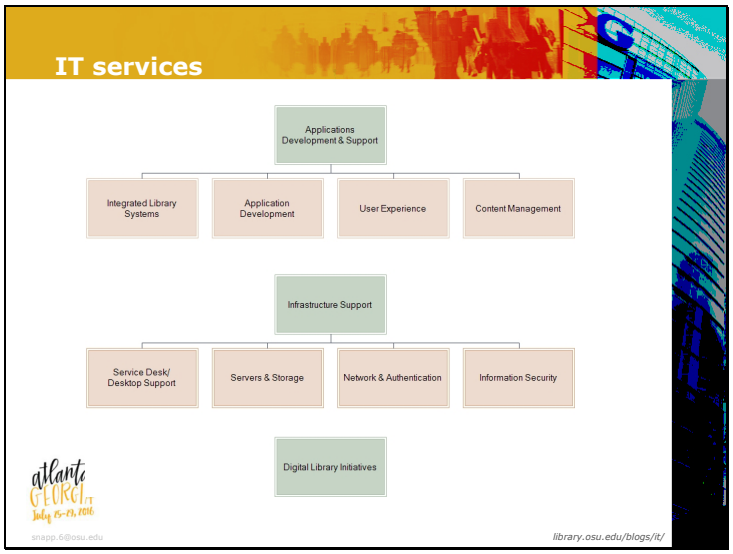
Slide 14

This is an example of a tear sheet from a newspaper from 1904.

Slide 15

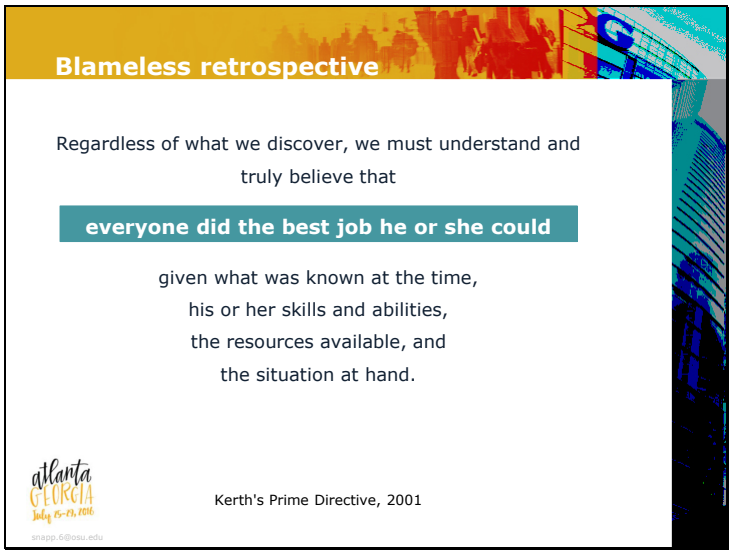
You can zoom in and see the tiniest of details which can be fun. So we're giving researchers access to materials that have previously only been available at the physical site. And making the materials available online to anyone in the world who might be interested, not just researchers.

Slide 16



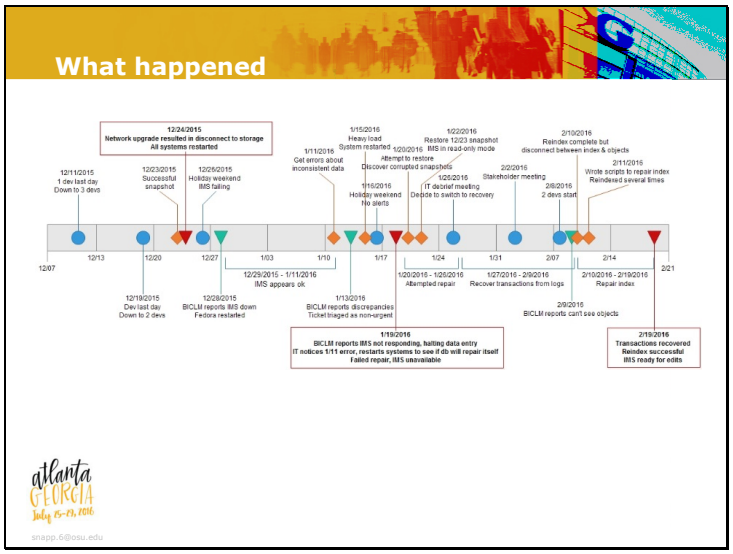
So that's our project. What about our team? We're a small team of 17 people. And you can see structurally there are 3 departments—Infrastructure Support, Digital Initiatives (product manager), and Applications Development & Support. There are 5 developers on my team when we are at full staff—challenging to retain developers right now with the competition in the Columbus market right now. And 2 systems administrators in Infrastructure. We practice agile software development, Scrum for new development and Kanban for maintenance projects. And, we have adopted many of the tools that would be considered part of the DevOps toolchain such as Puppet, Nagios, Splunk, Capistrano, etc.

Slide 17



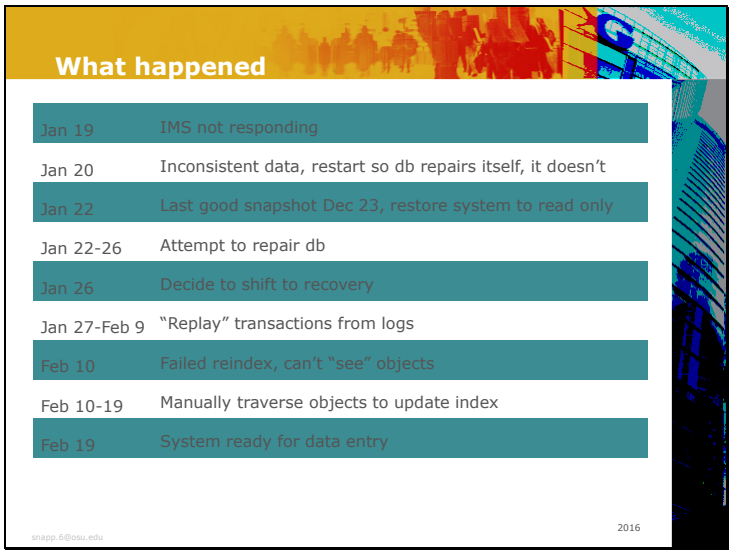
Before I talk about the events leading up to the disaster, I would like to pause a second and remind everyone of Kerth's Prime Directive for project retrospectives. You may think that this would never happen to your team. We did make some mistakes, but I want you to know that everyone did the best job he or she could under the circumstances, at the time, and I ultimately am accountable.

Slide 18

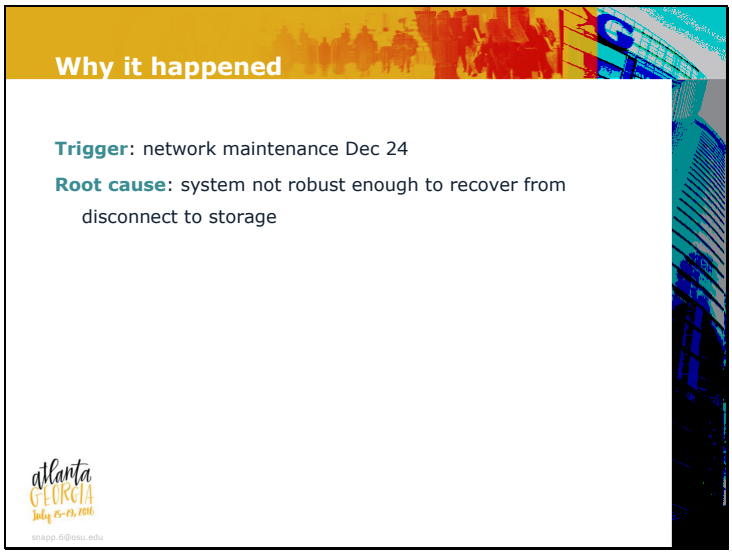


I'm not expecting you to be able to read the timeline, and I don't want to get too down in the weeds. But I think it's important to appreciate the time frame and complexities of the events at some level. The entire series of events happened between Dec 24 through Feb 19, 2016.

Slide 19

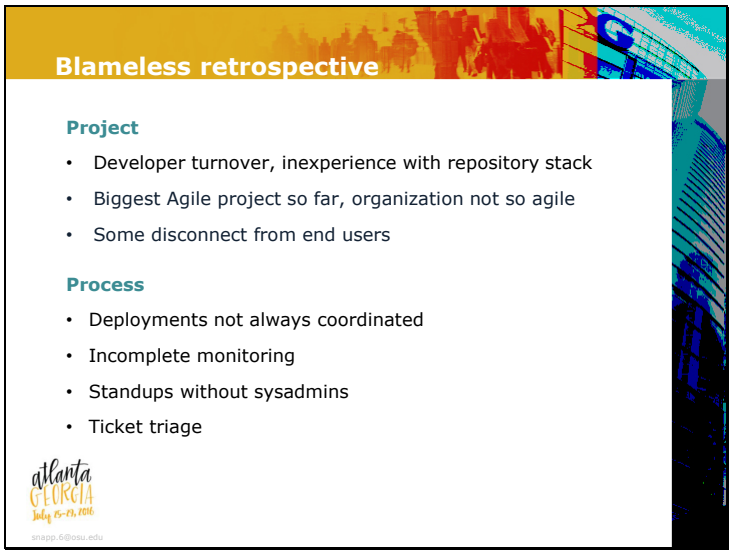


The critical event was on January 19 when the customer reported that IMS was not responding and they were stopping data entry—they were frustrated and had enough with the system. That's when we realized something serious was going on and went back into the logs and saw there were errors on Jan 11 about inconsistent data. So there's this thing called LevelDB, key value store, very low in the stack that keeps track of the pointers to where the objects are physically stored. We restarted the system expecting LevelDB to repair itself. It didn't. IMS goes down. We shift to restoring on Jan 22. That's when we discovered that the snapshots were corrupted since Dec 24. The last good snapshot was Dec. 23. We go ahead and restore that snapshot to get the system back up in read only mode to buy us some time to figure out what was going on with the system. We spent a week Jan 22-26, trying to repair LevelDB. That wasn't happening, despite reaching out to several experts in the community, so we shifted to recovering the data that had been entered since the last good snapshot. Our devs wrote scripts to essentially replay transactions from our different logs. Thankfully not that many. On Feb 10, the reindex finished but the customer reported, what they described as, not being able to "see anything". The problem was that the index and objects were out of sync and permissions had reverted to the most restrictive. We tried to reindex several more times. Finally, our lead dev wrote scripts to traverse the tree of objects and manually repair the index. On Feb 19, the system was finally ready for the customer to start uploading new objects. That's 3 days of the system being completely unavailable to anyone, and many intermittent outages as we reindexed. And one month of downtime for the admins who needed to ingest images.



To summarize, there was a network upgrade during regular maintenance on Dec 24 that resulted in an interruption between systems and storage. All of our systems were restarted, and every system we have gracefully came back up. Except one--Fedora--and unfortunately, we didn't discover that until much later. So the trigger was a network drop, and ultimately, the root cause was that Fedora was not robust enough to recover from a disconnect to storage. Those are the technical causes, but there were a lot of other things going on.

Slide 22



During the retrospective with the project team, including devs, my boss, my peers, and sysadmins, we identified other factors related to personnel, the project, and the process.

PERSONNEL

- No single person had the full picture: each person on the team had a piece of the puzzle and it took a long time to put the puzzle together. One reason was that we had experienced a lot of turnover in developers. We lost 7 devs in 3 years including the original lead developer on the project and the second lead. One of our sysadmins temporarily left the Libraries. So only one dev really understood the architecture. We were onboarding 2 new devs at the tail end of the disaster. That's a lot of churn in personnel.

PROJECT

- Other than for the ScrumMaster, we had limited face-to-face contact with the product owner and end users. It was the biggest agile project we had taken on up to that point, and the Libraries was not ready yet for Agile, culturally. I should also mention that Fedora was supposed to be a "black box", written in Java. Our devs are Ruby developers who were hired to develop the application.

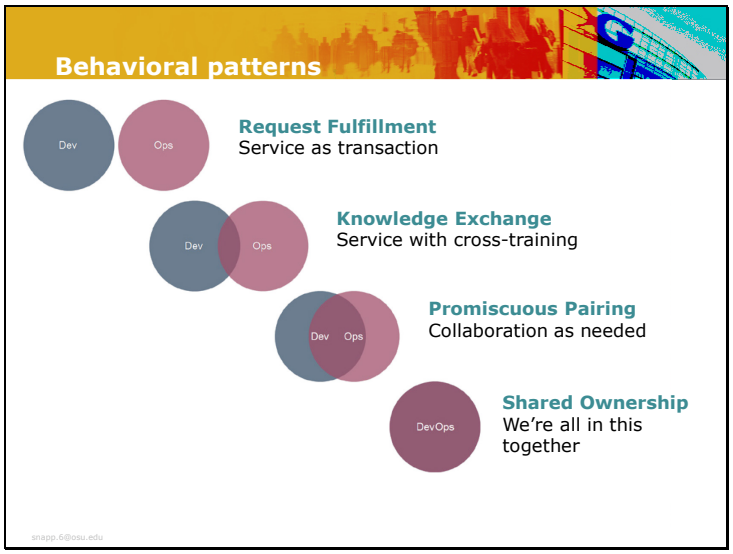
PROCESS

- We experienced several breakdowns in process, especially with alerting and monitoring. Roles were not always clear about who is supposed to respond to what alerts. Jira, our ticketing system, tended to be noisy at the time and we didn't always do a great job triaging and responding quickly. And that one critical ticket slipped through the cracks.

Slide 23

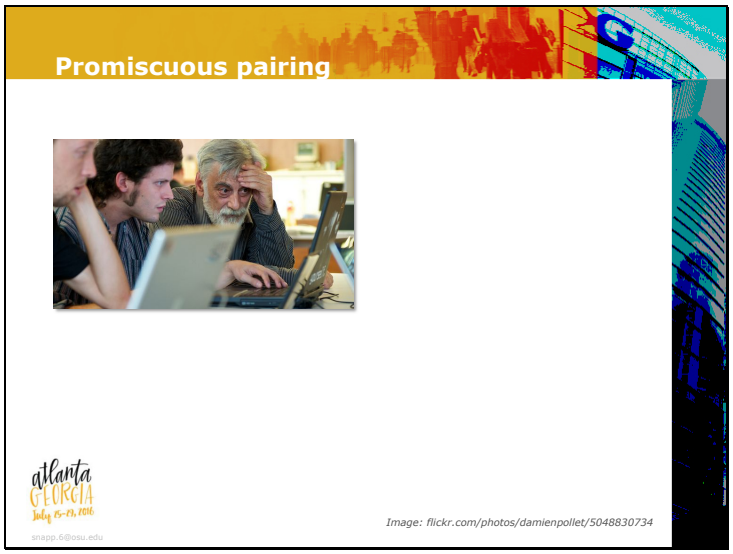
At this point, I had a bit of a nervous breakdown. I was totally deflated as a manager. From my perspective, we had positive working relationships. There was no interpersonal conflict that I observed with lots of productive chatter and joking in the DevOps HipChat room. I wasn't seeing the stereotypical silos between developers and infrastructure which I had experienced in past jobs. We had adopted DevOps tools. We are DevOps! So there clearly was a disconnect between my perception and the reality of the situation which led me to think a lot about the nature of the relationship between developers and sysadmins before, during and after the disaster. I could sense in the retrospective that the sysadmins were frustrated and the devs were pretty much clueless about what was going on. What was that really about?

Slide 26



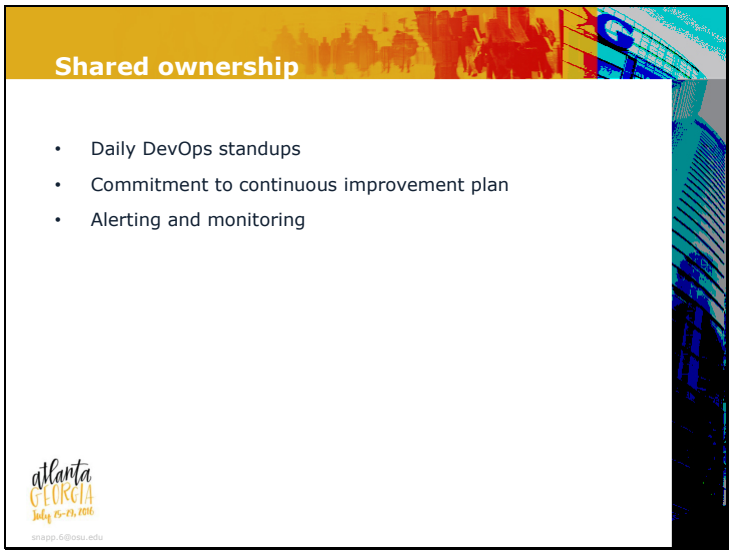
I came up with this model to represent the behavioral patterns I was observing over the course of the disaster. I gave you a couple examples of a transactional relationship between devs and ops which I would describe as (ITIL) request fulfillment. There were 3 other patterns (knowledge exchange, promiscuous pairing, and shared ownership) during the disaster, and we waffled back and forth and back again into these patterns.

Slide 30



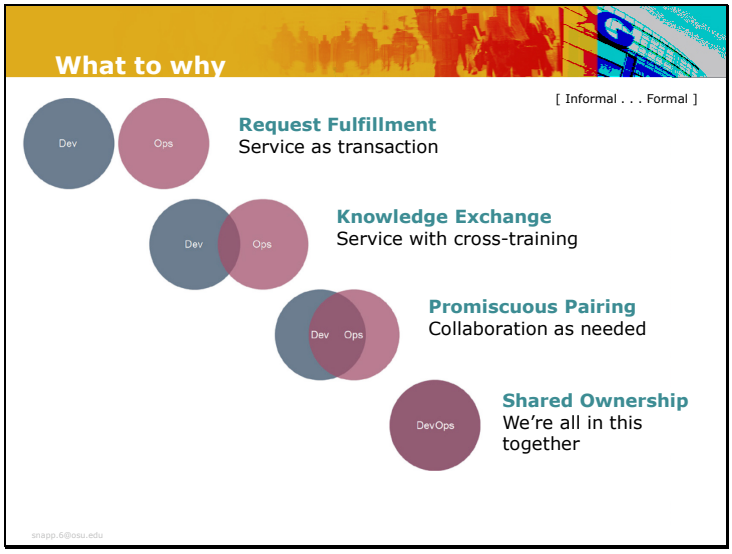
Developers pair regularly on our team, and during the disaster, developers and sysadmins paired. I would see them sitting together and working heads-down to solve problems. Collaboration was as needed and targeted toward a single problem.

Slide 31



How did we express shared ownership? “We’re all in this together.” We had daily standups with the project team during the disaster for updates and problem-solving. We developed a continuous improvement plan before we met with the stakeholders to demonstrate how we were going to prevent the event from happening again. We as a team were committed to that plan, because we had recognized failures in our processes, like monitoring and alerting. I work very well with my peer in Infrastructure, and we worked to bring our teams together during the disaster. These were moments when I felt that both teams were in it together.

Slide 33

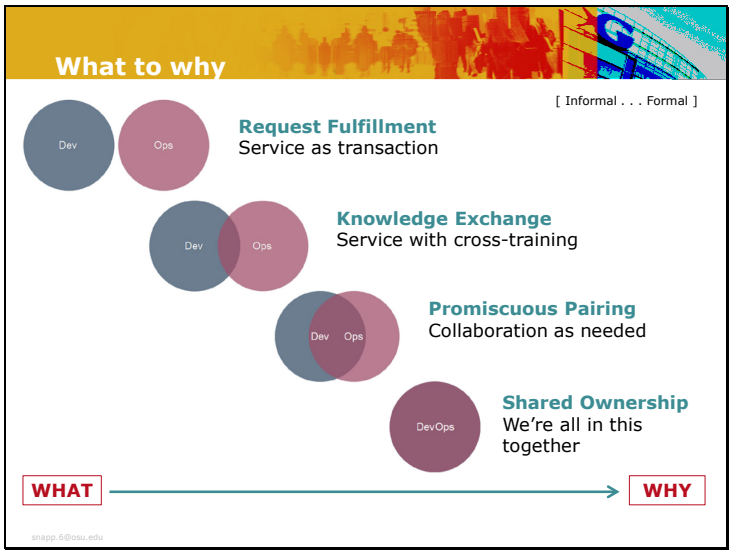


We could add another dimension to this model to indicate whether these are formal or informal arrangements (request fulfillment: chat vs tickets; knowledge exchange: could have formal code reviews). I would argue that request fulfillment doesn't reflect a DevOps culture when devs and sysadmins interact through transactions without any context or learning. So the disaster in my opinion brought us together and pushed us forward toward a true DevOps culture. And if you think of the patterns along a continuum, I would like to take a leap here and suggest that you could link these patterns to Sinek's Golden Circle.

Slide 34

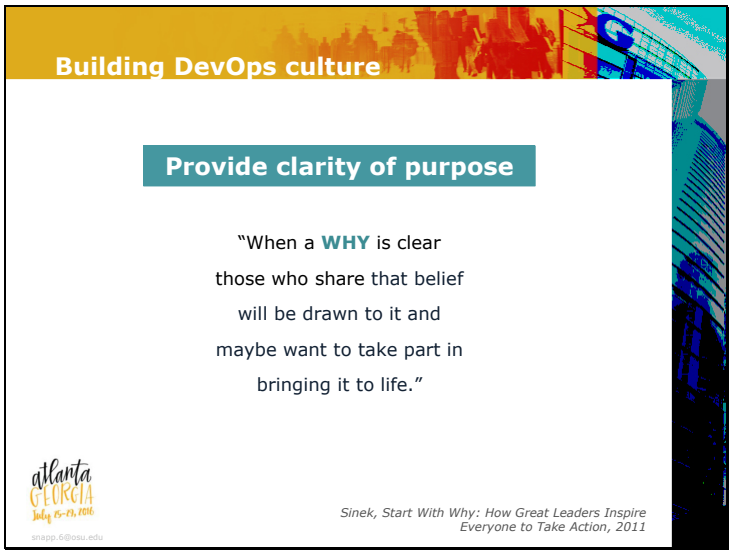
Sinek talks about leadership and successful organizations in terms of what, how, why and he suggests that the most successful companies like Apple are successful because they clearly articulate their purpose--we buy WHY they do what they do, not WHAT they make. Sinek says if you want to be successful, you need to "start from WHY".

Slide 35



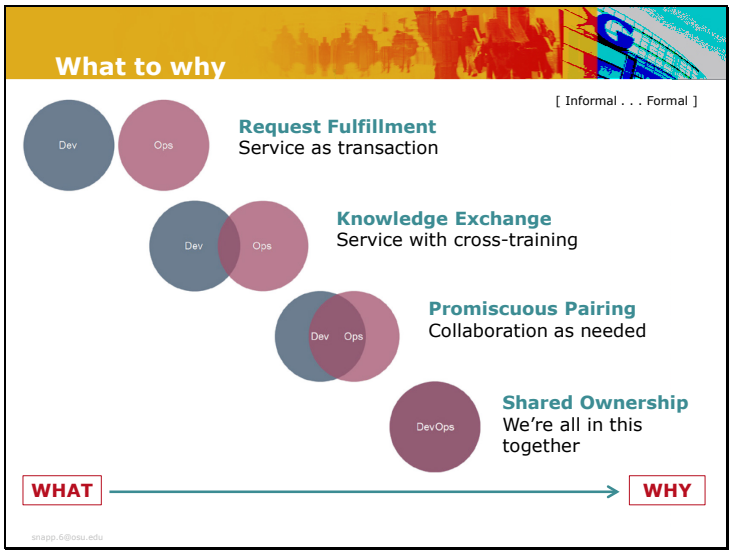
Perhaps we could link these patterns to WHAT and WHY. When your team is clear on WHY they're coming to work each day, they share ownership of their success, which to me is really what DevOps culture is about. How does this relate to our own experience? A key turning point during the disaster was when both the lead developer and the lead sysadmin sat with the stakeholders during a very uncomfortable and stressful debrief meeting. I described what had happened and how we were going to fix it and how we were going to prevent it from happening again. One of the end users became emotional—tears in her eyes--when she described her frustration with the system. She couldn't do her job. She no longer had trust in the system or in us. That hurt a lot to hear. That moment reminded me of our purpose: WHY do what we do. We are here to make our customers' jobs easier and we were failing.

Slide 36



So I believe that the single most important thing a manager can do to build a DevOps culture is to provide clarity of purpose. Everything else such as the automation tools will fall in place from there in support of that purpose. For us, our WHY is: anyone in the world can view cartoons from the BICLM, our HOW: DevOps tools and workflows; and our WHAT: build and release software.

Slide 37



Do you have examples of these patterns on your own teams?

Slide 38

What can we do to sustain a shared ownership culture? Here are some ideas.

Slide 39

I would like to express appreciation to everyone on the project teams.

Slide 40


References & Resources
Allspaw & Hammond, "10 deploys per day: Dev and ops cooperation at Flickr," Velocity, 2009
Babcock, "DevOps: A Culture Shift, Not A Technology," InformationWeek, 2014
Davis & Daniels, Effective DevOps: Building a Culture of Collaboration, Affinity, and Tooling at Scale, O'Reilly Media, 2016
Debois, "Devops Areas - Codifying devops practices," Jedi Blog, 2012
Hulme, "A Q & A with Gene Kim: DevOps and What Separates High Performing Enterprises," devops.com, 2014
Humble, "DevOps and Agile Release Management," Agile Australia, 2010
Kerth, Project Retrospectives: A Handbook for Team Reviews, Dorset House, 2001
Kim, Behr & Spafford, The Phoenix Project: A Novel about IT, DevOps, and Helping Your Business Win, IT Revolution Press, 2014
Mueller, "What Is DevOps?", the agile admin, revised 2016
Sinek, "How Great Leaders Inspire Action," TED2009
Sinek, Start With Why: How Great Leaders Inspire Everyone to Take Action, Portfolio, 2011
Sinek, "Why Good Leaders Make You Feel Safe," TED2014
Sinek, Leaders Eat Last: Why Some Teams Pull Together and Others Don't, Portfolio, 2014
Walls, Building a DevOps Culture, O'Reilly Media, 2013
Willis, "What Devops Means to Me," Chef Blog, 2010
Willis, "The Convergence of DevOps," IT Revolution Press, 2012
Zoltak & Narayanaswamy, "DevOps, Common use cases, Architectures, Best Practices," Amazon Web Services, 2015
atlanta GEORGIA
July 15-19, 2016
snapp.6@osu.edu